Robust Eye Centers Localization with Zero–Crossing Encoded Image Projections

Laura Florea LAURA.FLOREA@UPB.RO
Image Processing and Analysis Laboratory
University "Politehnica" of Bucharest, Romania, Address Splaiul Independenţei 313

Corneliu Florea CORNELIU.FLOREA@UPB.RO
Image Processing and Analysis Laboratory
University "Politehnica" of Bucharest, Romania, Address Splaiul Independenţei 313

Constantin Vertan CONSTANTIN.VERTAN@UPB.RO
Image Processing and Analysis Laboratory
University "Politehnica" of Bucharest, Romania, Address Splaiul Independenţei 313

Abstract

This paper proposes a new framework for the eye centers localization by the joint use of encoding of normalized image projections and a Multi Layer Perceptron (MLP) classifier. The encoding is novel and it consists in identifying the zero-crossings and extracting the relevant parameters from the resulting modes. The compressed normalized projections produce feature descriptors that are inputs to a properly-trained MLP, for discriminating among various categories of image regions. The proposed framework forms a fast and reliable system for the eye centers localization, especially in the context of face expression analysis in unconstrained environments. We successfully test the proposed method on a wide variety of databases including BioID, Cohn-Kanade, Extended Yale B and Labelled Faces in the Wild (LFW) databases.

1. Introduction

As noted in the review on the eye localization topic, "eye detection and tracking remains challenging due to the individuality of eyes, occlusion, variability in scale, location, and light conditions"(Hansen & Ji, 2010). Eye data and details of eye movements have numerous applications in face detection, biometric identification, and particularly in human-computer interaction tasks. Among the various applications of the eye localization topic, we are particularly interested in face expression analysis. Thus, while any method is supposed to perform accurately enough on the real-life cases and be fast-enough for real-time applications, we show an additional interest in the cases where eye centers are challenged by face expression. We will prove that the proposed method, which uses a MLP to discriminate among encoded normalized image projections from patches centered on the eye and, respectively, from patches shifted from the eye, is both accurate and fast.

1.1. Related work

The problem of eye localization was well investigated in literature, within a long history (Hansen & Ji, 2010). Methods for eye center (or iris or pupil) localization in passive, remote imaging may approach the problem either as a particular case of pattern recognition application, (Hamouz et al., 2005), (Asteriadis et al., 2009) or by using the physical particularities of the eye, like the high contrast to the neighboring skin (Wu & Zhou, 2003) or the circular shape of the iris (Valenti & Gevers, 2008). The proposed method combine a pattern recognition approach with features that make use of the eye's high contrast.

One of the first eye localization attempts is in the work from (Kanade, 1973), who used image projections for this purpose. Taking into account that our method also uses image projections for localization, in the next paragraphs we shall present state of the art methods by going from the conceptually closest to the wider categories. Namely we shall start by presenting solutions based on projections, to follow with general eye localization methods and face fiducial points localization algorithms.

As a general observation, we note that while older solutions (Jesorsky et al., 2000), (Wu & Zhou, 2003), tried to estimate also the face position, since the appearance of the Viola-Jones face detection solution (Viola & Jones, 2004), eye center search is limited to a subarea within the upper face square.

Projections based methods The same image projections as in the work of Kanade are used to extract information for eye localization in a plethora of methods (Feng & Yuen, 1998), (Zhou & Geng, 2004), (Turkan et al., 2004). (Feng & Yuen, 1998) start with a snake based head localization followed by anthropometric reduction (relying on the measurements from (Verjak & Stephancic, 1994)) to the so-called eye-images and introduce the variance projections for localization. The key points of the eye model are the projections particular values, while the conditions are manually crafted.

(Zhou & Geng, 2004) describe convex combinations between integral image projections and variance projections that are named generalized projection functions. These are filtered and analyzed for determining the center of the eye. The analysis is also manually crafted and requires identification of minima and maxima on the computed projection functions. Yet in specific conditions, such as intense expression or side illumination, the eye center does not correspond to a minima or a maxima in the projection functions. (Liu et al., 2010) use similar conditions with the ones used in (Zhou & Geng, 2004) but applied solely on the integral projections to detect if an eye is open or closed.

2004 introduce the edge projections and use them to roughly determine the eye position. Given the eye region, a feature is computed by concatenation of the horizontal and vertical edge image projections. Subsequently, a SVM–based identification of the region with the highest probability is used for marking the eye. The method from (Turkan et al., 2004) is, to our best knowledge, the only one, except ours which uses image projections coupled with machine learning. Yet we differ by using supplementary data coupled with the introduction of efficient computation techniques and elaborated pre and post-processing steps to keep the accuracy high and the running time low.

General eye localization methods There are many other approaches to the problem of eye localization. 2000 propose a face matching method based on the Hausdorff distance followed by a MLP eye finder. 2003 even reversed the order of the typical procedure: they use eye contrast specific measures to validate possible face candidates.

2004 rely on the Pairwise Reinforcement of Feature Responses algorithm for feature localization. 2006 use SVM on optimally selected Haar wavelet coefficients.

2005 refine with SVM the Gabor filtered faces, for locating 10 points of interest; yet the overall approach is different from the face feature fiducial points approach that is discussed in the next paragraph. 2006 use an iteratively bootstrapped boosted cascade of classifiers based on Haar wavelets. 2007 use multi scale Gabor jets to construct an Eye Model Bunch. 2009 use the distance to the closest edge to describe the eye area. 2008, (Valenti & Gevers, 2012) use isophote's properties to gain invariance and follow with subsequent filtering with Mean Shift (MS) or nearest neighbor on SIFT feature representation for higher accuracy. 2010 relies on thresholding the cumulative histogram for segmenting the eyes. 2010 train a set of classifiers to detect multiple face landmarks, including explicitly the pupil center, by using a sliding window approach and test in all possible locations and interconnect them to estimate the shape overall. 2011 rely their eye localizer on gradient techniques and search for circular shapes. 2013 use an exhaustive set of similarity measures over basic features such as histograms, projections or contours to extract the eye center location having in mind the specific scenario of driver assistance.

Face fiducial points localization More recently, motivated by the introduction by 2001 of the active appearance models (AAM), that simultaneously determine a multitude of face feature points, a new class of solutions, namely the localization of face fiducial points appeared. In this category we include the algorithm of 2005 who use a GentleBoost algorithm for combining Gabor filters extracted features; 2008, who extend the original active shape models with more landmark points and stacks two such models; 2012, who model shapes using the Markov Random Field and classify them using SVM in the so-called Borman algorithm; 2011 who use Bayesian inference on SIFT extracted features and most recently 2012 who use a combination of regularized boosted classifiers and mixture of complex Bingham distributions over texture and shape related features.

1.2. Paper Structure

In this paper we propose a system for eye centers localization that starts with face detection and illumination type detection, followed by a novel feature extractor, a MLP classifier for discriminating among possible candidates and a post-processing step that determines the eye centers. We contribute by:

- Describing a procedure for *fast* image projections computation. This step is critical in having the solution run in real time.
- Introducing a *new encoding* technique to image analysis domain.
- The combination of *normalized* image projections with zero-crossing based encoding results in image description features named *Zero-crossing based Encoded image Projections* (ZEP). They are fast, simple, robust and easy to compute and therefore have applicability in a wider variety of problems.
- The integration of the features in a *framework* for the problem of eye localization. We will show that description of the eye area using ZEP leads to significantly better results than state of the art methods in real-life cases represented by the extensive and very difficult Labeled Faces in the Wild database. Furthermore, the complete system is the fastest known in literature among the ones reporting high performance.

The remainder of this paper is organized as follows: Section 2 reviews the concepts related to Integral Projections and describes a fast computation method for them; Section 3 summarizes the encoding procedure and the combination with image projections to form the ZEP features. The paper ends with implementation details, with a discussion on the achieved results in the field of eye localization and proposals for further developments.

2. Image Projections

2.1. Integral Image Projections

The integral projections, also named integral projection functions (IPF) or amplitude projections, are tools that have been previously used in face analysis. They appeared as "amplitude projections" (Becker et al., 11) or as "integral projections" (Kanade, 1973) for face recognition. For a gray-level image sub–window $I(i,j)$ with $i = i_m \dots i_M$ and $j = j_n \dots j_N$, the projection on the horizontal axis is the average gray–level along the columns (1), while the vertical axis projection is the average gray–level along the rows (2):

$$P_H(j) = \frac{1}{i_M - i_m + 1} \sum_{i=i_m}^{i_M} I(i,j), \forall j = j_n, \dots, j_N \tag{1}$$

$$P_V(i) = \frac{1}{j_N - j_n + 1} \sum_{j=j_n}^{j_N} I(i,j), \forall i = i_m, \dots, i_M \tag{2}$$

The integral projections reduce the dimensionality of data from 2D to 1D, describing it up to a certain level of details. Also, the projections can be computed on any orthogonal pair of axes, not necessarily rows and columns. This will be further discussed in subsection 5.2.

2.2. Edge Projections

Over time, several extensions of the integral projections have been introduced such as variance projection functions (Feng & Yuen, 1998) or edge projection functions (EPF) (Turkan et al., 2004).

Instead of determining edges with wavelet transform as in the case of (Turkan et al., 2004), we use a different approach for computing the edge projections. First, the classical horizontal and vertical Sobel contour operators (for details see (Gonzalez & Woods, 2001) sect. 3.7) are applied, resulting in S_H and S_V which are combined in the $S(i,j)$ image used to extract edges:

$$S(i,j) = S_H^2(i,j) + S_V^2(i,j) \tag{3}$$

The edge projections are computed on the corresponding image rectangle $I(i,j)$:

$$E_H(j) = \frac{1}{i_M - i_m + 1} \sum_{i=i_m}^{i_M} S(i,j), \forall j = j_n, \dots, j_N \tag{4}$$

$$E_V(i) = \frac{1}{j_N - j_n + 1} \sum_{j=j_n}^{j_N} S(i,j), \forall i = i_m, \dots, i_M \tag{5}$$

Equations (4) and (5) are simply equations (1) and (2) applied on the Sobel edge image $S(i,j)$.

As Sobel operator is invariant to additive changes, if compared to other types of projections, the edge projections are significantly more stable with respect to illumination changes.

2.3. Fast Computation of Projections

While sums over rectangular image sub–windows may be easily computed using the concept of summed area tables (Crow, 1984) or integral image (Viola & Jones, 2004), a fast computation of the integral image projections may be achieved using the *prefix sums* (Blelloch, 1990) on rows and respectively on columns. A prefix-sum is a cumulative array, where each element is the sum of all elements to the left of it, inclusive, in the original array. They are the 1D equivalent of the integral image, but they definitely precede it as recurrence $x_i = a_i + x_{i-1}$ is known for many years.

For the fast computation of image projections, two tables are required: one will hold prefix sums on rows (a table which, for keeping the analogy with integral image, will be named horizontal 1D integral image) and respectively one vertical 1D integral image that will contain the prefix sums on columns. It should be noted that computation on each row/column is perform separately. Thus, if the image has $M \times N$ pixels, the 1D horizontal integral image, on the column j, $\mathcal{I}_H^j$, is:

$$\mathcal{I}_H^j(i) = \sum_{k=1}^{i} I(k,j) \ \ , \forall i = 1, \dots M \tag{6}$$

Thus, the horizontal integral projection corresponding to the rectangle $i = [i_m; i_M] \times [j_n; j_N]$ is:

$$P_H(j) = \frac{1}{i_M - i_m + 1}\left(\mathcal{I}_H^j(i_M) - \mathcal{I}_H^j(i_m - 1)\right) \tag{7}$$

The procedure is visually exemplified in figure 1.

Using the oriented integral images, the determination of the integral projections functions on all sub-windows of size $K \times L$ in an image of $M \times N$ pixels requires one pass through the image and $2 \times M \times N$ additions, $2 \times (M - K) \times (N - L)$ subtractions and two circular buffers of $(K + 1) \times (N + 1)$ locations, while the classical determination requires $2 \times K \times L \times (M - K) \times (N - L)$ additions. Hence, the time to extract the projections associated with a sub-window, where many sub-windows are considered in an image, is greatly reduced.

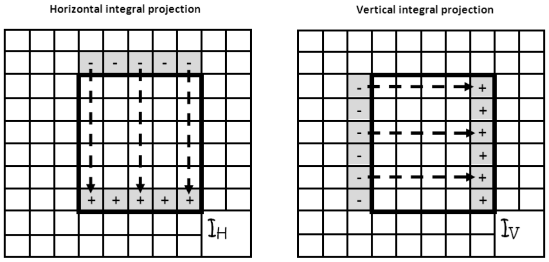


Figure 1. Given the two 1D oriented integral images, $\mathcal{I}_H^j$ and $\mathcal{I}_V^i$, each element of the integral projections, P_H and P_V that describes the marked sub-window is found by a simple subtraction.

The edge projections require the computation of the oriented integral images over the Sobel edge image, $S(i,j)$. This image needs to be computed on the areas of interest.

In conclusion, the fast computation of projections opens the direction of real-time feature localization on high resolution images.

3. Encoding and ZEP Feature

To reduce the complexity (and computation time), the projections are compressed using a zero-crossing based encoding technique. After ensuring that the projections values are in a symmetrical range with respect to zero, we will describe, independently, each interval between two consecutive zero-crossings. Such an interval is called an *epoch* and for its description three parameters are considered (as presented in figure 2):

- *Duration* - the number of samples in the epoch;
- *Amplitude* - the maximal signed deviation of the signal with respect to 0;
- *Shape* - the number of local extremes in the epoch.

The proposed encoding is similar with the TESPAR (Time-Encoded Signal Processing and Recognition) technique (King & Phipps, 1999) that is used in the representation and recognition of 1D, band–limited, speech signals. Depending on the problem specifics, additional parameters of the epochs may be considered (e.g. the difference between the highest and the lowest mode from the given epoch). Further extensions are at hand if an epoch is considered the approximation of a probability density function and the extracted parameters are the statistical moments of the said distribution. In such a case the *shape* parameter corresponds to the number of modes of the distribution.

The reason for choosing this specific encoding is twofold. First the determination of the zero-crossings and

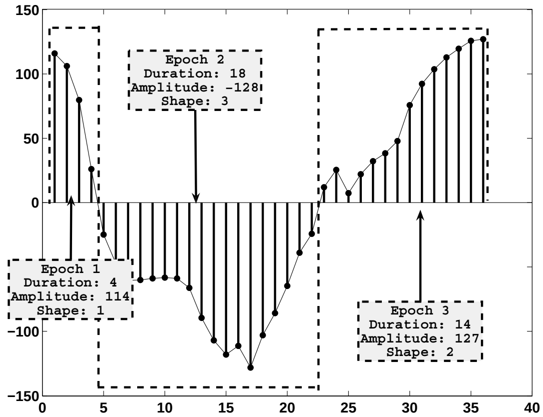


Figure 2. Example of 1D signal (vertical projection of an eye crop) and the associated encoding. There are three epochs, each encoded with three parameters. The associated code is: $[4, 114, 1; \quad 18, -128, 3; \quad 14, 127, 2]$

the computation of the parameters is doable in a single pass through the target 1D signal, and, secondly, the epochs have specific meaning when describing the eye region, as discussed in the next subsection.

Given an image sub-window, the ZEP feature is determined by the concatenation of four encoded projections as described in the following:

1. Compute both the integral and the edge projection functions (P_H, P_V, E_H, E_V);

2. Independently *normalize* each projection within a symmetrical interval. For instance, in our application we normalized each of the projections to the $[-128; 127]$ interval. This will normalize the amplitude of the projection;

3. Encode each projection as described; allocate for each projection a maximum number of epochs;

4. Normalize all other (i.e. duration and shape) encoding parameters;

5. Form the final Zero-crossing based Encoded image Projections (ZEP) feature by concatenation of the encoded projections. Given an image rectangle, the ZEP feature consists of the epochs from all the 4 projections: (P_H, P_V, E_H, E_V).

Image projections are simplified representations of the original image, each of them carrying specific information; the encoding simplifies even more the image representation. The normalization of the image projections, and thus of the epochs amplitudes, ensures independence of the ZEP feature with respect to uniform variation of the illumination. The normalization with respect to the number of elements in the image sub-window leads to partial scale invariance: horizontal projections are invariant to stretching on the vertical direction and vice versa. The scale invariance property of the ZEP feature is achieved by completely normalizing the encoded durations to a specific range (e.g. the encoded horizontal projection becomes invariant to horizontal stretching after duration normalization). We stress that when compared with previous methods based on projections, which lack the normalization steps, the hereby proposed algorithm increases the overall stability to various influences.

3.1. ZEP on Eye Localization

As noted, image projections have been used in multiple ways for the problem of eye localization. In an exploratory work, 1973 determined the potential of image projections for face description. More recently, 1998, 2004 and 2004 presented the use of the integral projections and/or their extensions for the specific task of eye localization. Especially in (Zhou & Geng, 2004) it was noted that image projections, in the eye region have a specific sequence of relative minima and maxima assigned with to skin (relative minimum), sclera (relative maximum), iris (relative minimum), etc.

Considering a rectangle from the eye region including the eyebrow (as showed in figure 3 (a)), the associated integral projections have specific epochs, as showed in figure 3 (c) and (d). The particular succession of positive and negative modes is precisely encoded by the proposed technique. On the horizontal integral projection there will be a large (one-mode) epoch that is assigned to skin, followed by an epoch for sclera, a triple mode, negative, epoch corresponding to the eye center and another positive epoch for the sclera and skin. On the vertical integral projection, one expects a positive epoch above the eyebrow, followed by a negative epoch on the eyebrow, a positive epoch between the eyebrow and eye, a negative epoch (with three modes) on the eye and a positive epoch below the eye.

The ZEP feature, due to invariance properties already discussed, achieves consistent performance under various stresses and is able to discriminate among eyes (patches centered on pupil) and non-eyes (patches centered on locations at a distance from the pupil center). As explained in section 4.2, on the validation set, using Fisher linear discriminant over 90% correct eye detection rate is achieved by selecting patches that are centered on the pupil with respect to the ones that are shifted.

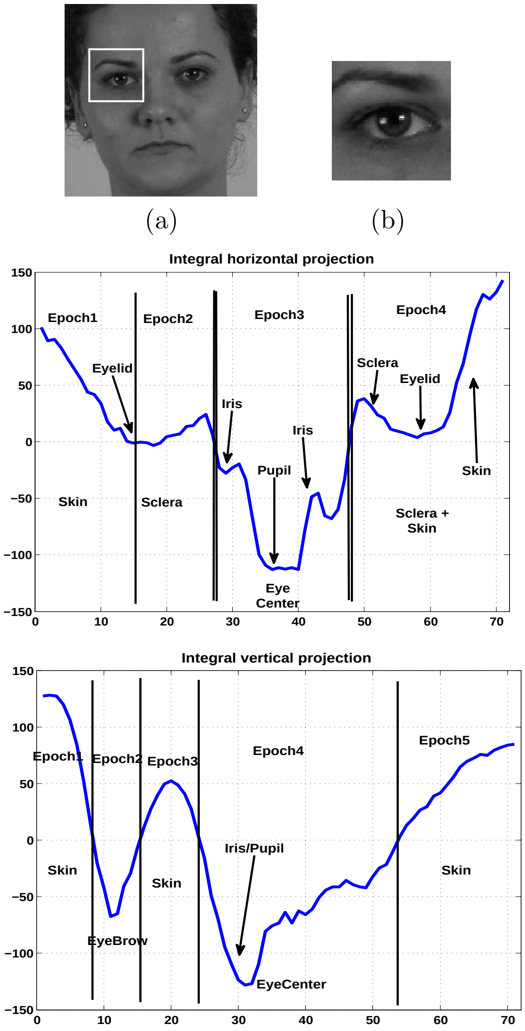

(a) (b) (c) (d)

Figure 3. Image projections from a typical eye patch: (a) face crop, (b) eye crop, (c) integral horizontal image projection on the eye crop, (d) integral vertical image projection on the eye crop. On the horizontal projection the double line marks the zero crossing that is found on all eye examples, while the rest of zero-crossings may be absent in some particular cases.

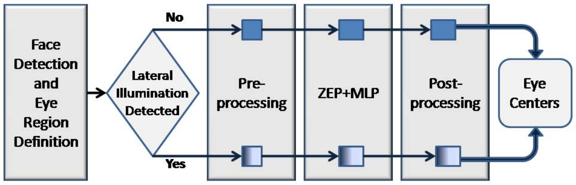

Face Detection and Eye Region Definition
Lateral Illumination Detected
No
Yes
Pre-processing
ZEP+MLP
Post-processing
Eye Centers

Figure 4. The work flow of the proposed algorithm.

4. Implementation

The block schematic of our eye center localization algorithm is summarized in figure 4. In the first step, a face detector (the cascade of Haar features (Viola & Jones, 2004) delivered with OpenCV) automatically determines the face square. Next the regions of interest are set in the upper third of the detected face: from 26% to 50% of the face square on rows, respectively from 25% to 37% on columns for the left eye and from 63% to 75% on columns for the right eye.

Noting the susceptibility of the image projections to alter their shape due to lateral illumination, we introduced a simple method for detecting such a case and we adapt the algorithm to the type of illumination found. After a very simple preprocessing, the ZEP features for each possible location are computed and feed to a classifier to identify the possible eye locations. The possible eyes are then post-processed and the best positions are located as discussed in subsection 4.3.

Regarding the face detection, the recent solutions use multiple cascades for not only identification of the face rectangle, but also for determination of the in-plane (roll) and yaw (frontal/profile) angles of head. Such procedure follows Viola and Jones extension of the initial face detector work (Jones & Viola, 2003), (Ramirez & Fuentes, 2008). Thus, it is customary to limit the analysis of "frontal faces" to a maximum rotation of 30^0.

4.1. Lateral Illumination Detection

To increase the solution robustness to lateral illumination, we automatically separate such cases. The motivation for the split lies in the fact that side illumination significantly alters the shape of the projections in the eye region, thus decreasing the performance of the classification part.

The lateral illumination detection relies on computing the average values on the eye patch previously selected. The following ratios are considered:

Table 1. Percentage of frontal illumination detected cases on the Extended Yale B database.

Azimuth / Elevation	$\pm[40^0 : 130^0]$	$\pm[0^0 : 35^0]$
$-40^0 : 0^0$	26.11%	88.28%
$10^0 : 90^0$	17.26%	36.35%

$$L_{ratio} = \frac{\overline{L_{top}}}{\overline{L_{bot}}}; \quad R_{ratio} = \frac{\overline{R_{top}}}{\overline{R_{bot}}}; \quad H_{ratio} = \frac{\overline{L_{top}}+\overline{L_{bot}}}{\overline{R_{top}}+\overline{R_{bot}}}; \tag{8}$$

where ($\overline{L_{top}}$) and ($\overline{L_{bot}}$) are the average gray levels on the upper and lower halves of the left eye and ($\overline{R_{top}}$ and $\overline{R_{bot}}$) are their correspondents on the right eye.

The lateral illumination case is considered if any of the computed ratios, L_{ratio}, R_{ratio}, H_{ratio}, is outside the $[0.5; 1.75]$ range.

We designed this block such that an illumination that do not produce significant shadows on the eye region is detected as frontal, and as lateral otherwise. In terms of illumination angle, the cases with shadows on the eye region imply an absolute value of azimuth angle higher than 40^0 or an elevation angle value higher than 10^0. Negative elevation (light from below) with low azimuth value does not produce shadows on the eye region. The interval $[0.5; 1.75]$ mentioned above has been found by matching the mentioned cases with the ratios values on the training database.

Indeed 98.54% of the images from the BioID database are detected as frontal illuminated, while the results on Extended Yale B are presented in table 1. Extended Yale B database has images with various illumination angles as it will be discussed in section 5.5.

4.2. Training

Once the ZEP features are determined, the extracted data is feed into a Multi-Layer Perceptron (MLP), having one input layer, one hidden layer and one output layer, trained with the back-propagation algorithm. In our implementation, the number of neurons from the hidden layer is chosen to be half the size of the ZEP feature as it was empirically determined as a reasonable trade-off between performance (higher number of hidden neurons) and speed.

In the preferred implementation, each projection is encoded with 5 epochs, leading to 60 elements in the ZEP feature (and 60 inputs to the MLP). If more epochs are provided by projections (which is very unlikely for eye localization - less than 0.1% in the tested cases), the last ones are simply removed.

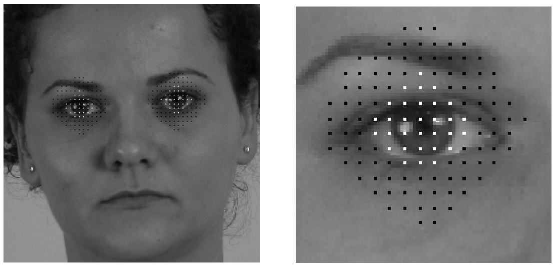


Figure 6. Localization of the image patch centers used as positive examples (white) and negative examples (black). The right hand image is zoomed from the left hand one.

The training of the MLP is performed with crops of eyes and non-eyes of 71 × 71 pixels, as shown in figure 5, while the preferred face size is 300 × 300 pixels. The positive examples are taken near the eye ground truth: the eye rectangle overlaps more than 75% with the true eye rectangle. The patches corresponding to the negative examples overlap with the true eye between 50% and 75%, thus leading to a total of 25 positive examples and 100 negative ones from each eye in a single face image. Positive and negative locations are showed in figure 6. This specific choice of positive and negative examples yields to high performance in localization.

In total there were 10,000 positive examples and as much negative ones, taken from the authors' Eye Chimera database, from the Georgia Tech database (Nefian & Hayes, 2000) and from the neutral poses selected from the YaleB database (Georghiades et al., 2001). We have considered two training variants corresponding to the two types of illumination (frontal or lateral).

One training procedure uses images from our data set (40%) and from Georgia Tech (60%) and focuses on frontal illumination, eye expressions and occlusions. In this case, the MLP is trained to return the L2 distance from a specific patch center to the true eye center. Thus the MLP performs regression.

The second training procedure (for lateral illumination) uses images only from the frontal pose of the Yale B database and it is used for improving performance against illumination. In this case the training set was labelled with -1 (non-eye – between 50% and 75% overlapping onto the centered eye patch) or 1 (eye – more than 75% overlapping).

As many machine learning algorithms are available, we have performed a short study on examples extracted from the training databases. Given the number of images in the databases, 20,000 examples were used for

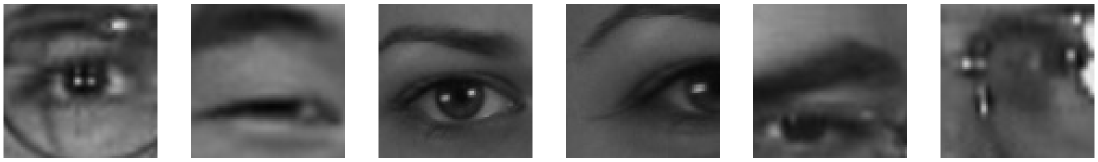


Figure 5. Examples of eye (first three images) and non-eye (last three images) crops; it should be noted that the examples include eye expressions and glasses

Table 2. Details of the two merged solutions of the actual eye localization

Case	**Frontal illumination**	**Lateral illumination**
Darkness preprocess. threshold	0.15	0.3
Training database	GeorgiaTech + Authors'	Yale B
Training scheme	Regression L2 dist	True/False
ZEP+MLP threshold	0	−0.5
eye area selection	largest lower region	largest region
eye center localization	weighted center of mass	geometrical center of the rectangle circumscribed to the eye region

training the networks and approximately 200,000 were used for classifier validation. For the classification problem, a Support Vector Machine (SVM) produced 93.7% correct detection rate, the used MLP 92.6% and an ensemble of 50 bagged decision trees 91.5% detection rate. For the regression case, a SVM for Regression lead to an approximation error of 0.090, the regression MLP 0.096 and bagged ensemble of regression trees only 0.115. Taking into account the achieved values, there is no significant performance difference among the various machine learning systems tested (conclusion which matches the findings from (Everingham & Zisserman, 2006)), thus our decision on using the MLP was based more on speed issues.

4.3. Preprocessing and Postprocessing

The conceptual steps in both illumination cases of the actual eye localization procedure are the same: preprocessing, machine learning and postprocessing.

A simple preprocessing is applied for each eye candidate region to accelerate the localization process. Following 2003, we note that the eye center (associated with the pupil) is significantly darker than the surrounding; thus the pixels that are too bright with respect to the eye region (and are not plausible to be eye centers) are discarded. The "too bright" characteristic is encoded as gray–levels higher than a percentage (so called darkness preprocessing threshold in table 2) from the maximum value of the eye region. In the lateral illumination case, this threshold is higher due to the deep shadows that can be found on the skin area surrounding the eye.

In the area of interest, using a step of 2 over a sliding image patch of 71 × 71 pixels, we investigate by the proposed ZEP+MLP all the plausible locations. We consider as positive results the locations where the value given by the MLP is higher than an experimentally found threshold (see table 2). These positive results are recorded in a separate image (the ZEP image, shown in figure 7) which is further post–processed for eye center extraction.

Since closed eyes (that were included in the training set) are similar with eyebrows, one may get false eye regions given by the eyebrow in the ZEP image. Thus the ZEP image is segmented, labelled and the lowest and largest regions are associated with the eye. This step will discard, for instance, the regions given by the eyebrow in figure 7 (c).

For the frontal illumination case, due to training with L2 distance as objective, one expects a symmetrical shape around the true eye center. Thus the final eye location is taken as the weighted center of mass of the previously selected eye regions. For the lateral illumination, the binary trained MLP is supposed to localize the area surrounding the eye center and the final eye center is the geometrical center of the rectangle circumscribed to the selected region. We note that in both cases, the specific way of selecting the final eye center is able to deal with holes (caused by reflections or glasses) in the eye region.

An overview of how each step is implemented in the two illumination cases considered is shown in table 2.

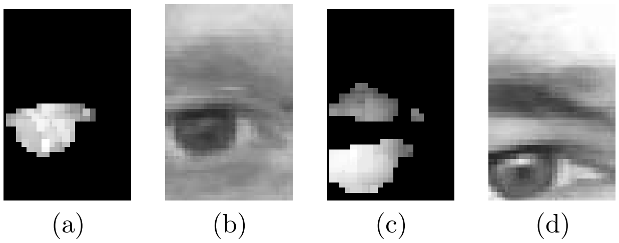


Figure 7. The ZEP image ((a) and (c)) resulting after processing with ZEP+MLP (so called ZEP image) the original eye patch image (b), (d) (under frontal illumination case). The first two images (a), (b) show a simple case (a) when only patches around eye center were selected by the MLP. The right hand two images, (c) and (d), contain a more difficult case (c) where also the eyebrow was identified as possible eye region. Higher gray–levels in the ZEP image signal higher confidence, as estimated by MLP, in having the eye center at that specific location.

5. Results and Discussions

We will discuss first the influence of various system parameters onto the overall results. For this purpose we will use the BioID database[1]. This database contains 1521 gray-scale, frontal facial images of size 384×286, acquired with frontal illumination conditions in a complex background. The database contains 16 tilted and rotated faces, people that wear eye-glasses and, in very few cases, people that have their eyes shut or pose various expressions. The database was released with annotations for iris centers. Being one of the first databases that provided facial annotations, BioID became the most used database for face landmarks localization accuracy tests, even that it provides limited variability and reduced resemblance with real-life cases. We will use BioID as a starting point in discussing the achieved results (for giving an inside on the system's various parameters and selecting the most performing state of the art systems) so that later to continue the evaluation under other stresses like eye expression, illumination angle or pose. Yet the most relevant test is on real-life cases, which are acquired in the Labelled Faces in Wild database that will be presented later on.

The localization performance is evaluated according to the stringent localization criterion (Jesorsky et al., 2000). The eyes are considered to be correctly determined if the specific localization error ϵ, defined in equation (9) is smaller than a predefined value.

$$\epsilon = \frac{\max\{\varepsilon_L, \varepsilon_R\}}{D_{eye}} \tag{9}$$

Table 3. Percentage of correct eye localization of the proposed algorithm (integral – IPF and edge projection functions – EPF) compared with IPF-only and EPF-only implementations

Projection Type	*IPF+EPF*	IPF	EPF
Accuracy, $\epsilon < 0.05$	70.46	56.61	53.17
Accuracy, $\epsilon < 0.10$	91.94	87.58	84.02
Accuracy, $\epsilon < 0.25$	98.87	98.81	96.50

In the equation above, ε_L is the Euclidean distance between the ground truth left eye center and determined left eye center, ε_R is the corresponding value for the right eye, while D_{eye} is the distance between the ground truth eyes centers. Typical error thresholds are $\epsilon = 0.05$ corresponding to eyes centers found inside the true pupils, $\epsilon = 0.1$ corresponding to eyes centers found inside the true irises, and $\epsilon = 0.25$ corresponding to eyes centers found inside the true sclera. This criterion identifies the worst case scenario.

We note that, while the BioID image size leads to approximately a 40×40 size for the eye patch, because our target are HD video frames (for which we will also provide duration), we upscale the face square to 300×300, thus having an eye square of 71×71.

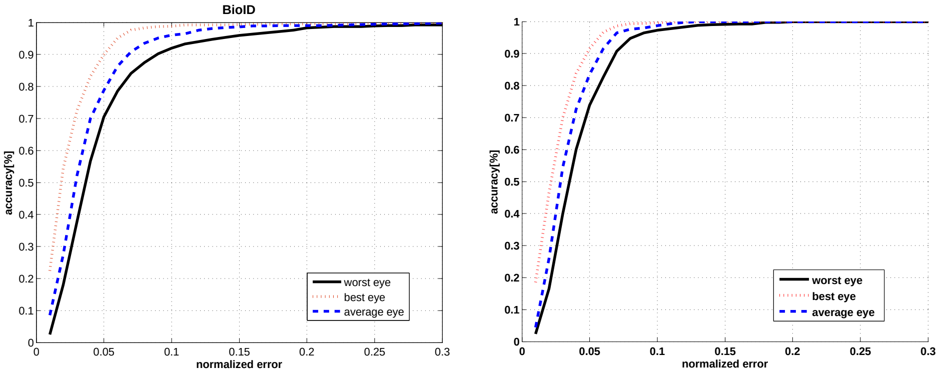
The results on the BioID database are shown in figure 8 (a), where we represented the maximum (better localized eye), average and minimum (worst localized eye) accuracy with respect to various values of the ε threshold.

5.1. The Influence of ZEP Parameters

We investigated the performance of the proposed system when only one type of projection is used. The results are presented in table 3. The computation time dropped to ≈ 53% of the full algorithm time if only one projection type is used. The performance drops with ≈ 14% in the case of integral projection and with ≈ 17% in the case of edge projections. Using the proposed encoding it is possible to keep both the dimensionality of the feature and the time duration low enough in order to use more than one type of projection. This supplementary information helps to increase the results accuracy when compared with the method in (Turkan et al., 2004).

Alternatives to the eye crop size and resulting values are presented in table 4. The experiment was performed by re-training the MLP with eye crops of the target size. As one can notice, the results are similar, thus proving the scale invariance of the ZEP feature. Slight variation is due to the pre- and post processing.

[1] http://www.bioid.com/downloads/software/bioid-face-database.html



(a)

(b)

Figure 8. Maximum (best case - red dotted line), average (blue dashed line) and minimum (worst eye - black solid line) accuracy for eye localization on the BioID database (left - (a)) and Cohn-Kanade database (right - (b)).

Table 4. The variation of the results with respect to the size of the eye analysis window.

Crop size	36×36	71×71	100×100
Accuracy, $\epsilon < 0.05$	64.00	70.46	64.20
Accuracy, $\epsilon < 0.10$	90.36	91.94	90.22
Accuracy, $\epsilon < 0.25$	98.61	98.87	99.14

Table 5. Percentage of correct eye localization of the proposed algorithm (ZEP encoding) compared with dimensionality reduction with PCA.

Encoding Type	*Proposed*	PCA	None
Accuracy, $\epsilon < 0.05$	70.46	69.66	72.97
Accuracy, $\epsilon < 0.10$	91.94	92.70	93.52
Accuracy, $\epsilon < 0.25$	98.89	98.87	99.07

5.2. The Dimensionality Reduction

The main visible effect of the proposed encoding is the reduction of the size of the concatenated projections. Yet, as we have adapted the encoding technique to the specific of the projection functions applied on the eye area, its performance is higher than of other methods. To see the influence of this encoding technique, we compared the achieved results with the ones obtained by reducing the dimensionality with PCA (as being the most foreknown such technique) by the same amount as the proposed one. The rest of the algorithm remains the same. The comparative results may be seen in table 5. We also report the results when no reduction was performed.

The results indicate that both methods are lossy compression techniques and lead to decreased accuracy. The proposed method is able to extract the specifics of the eye from the image projections, as discussed in subsection 3.1, being marginally better then the PCA compression.

Furthermore, we take into account that the dimensionality reduction with PCA requires, for each considered location, a matrix multiplication to project the initial vector of size N_p ($N_p = 284$) onto the final space (with size $M_p = 60$), thus having the complexity $\mathcal{O}(N_p M_p) = \mathcal{O}(284 \times 60)$. In comparison, the determination of the epochs parameters is done in a single cross of the initial vector (i.e. with complexity $\mathcal{O}(N_p) = \mathcal{O}(284)$), thus we expect the proposed method to be significantly faster.

Indeed, the average value for computation time increases from 6 msec (using the proposed method) to 11 msec (almost double) using PCA on a 300×300 face square. The lack of compression increases the duration to 24 msec per face square.

5.3. Robustness to Noise

An image projection represents a gray-scale average, hence it is reasonable to expect that the proposed method is very robust to noise. To study robustness to noise we have artificially added Gaussian noise to the BioID images and we subsequently measured the localization performance for $\epsilon < 0.1$ accuracy. Indeed, while the noise variance increases from 0 to 30, the average accuracy decreases from 91.94% to only 79.92%. The variation of the accuracy with respect to the added noise standard deviation may be seen in figure 9. Examples of images degraded by noise may be seen in figure 10.

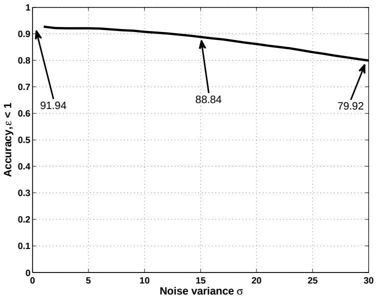
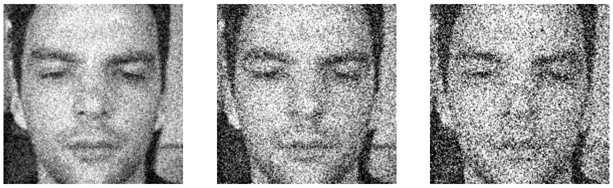


Figure 9. Variation of the localization performance for $\epsilon < 0.1$ accuracy with respect to gaussian noise variance used for image degradation.



$\sigma = 5$ $\sigma = 15$ $\sigma = 30$

Figure 10. Image from the BioID database degraded by noise. For a human observer it is quite difficult to determine the eye centers. Yet the results showed in figure 9 prove that our method performs remarkably well.

5.4. Results on the BioID

To give an initial overview of the problem in state of the art, we consider the results reported by other methods on the BioID database. Other solutions for eye localization are summarized in table 6. The results of the methods for localization of face fiducial points are showed in table 7. Visual examples of images with localized eyes produced by our method are shown in figure 11.

Analyzing the performance, first we note that our method significantly outperforms in both time and accuracy other methods relying on image projections ((Zhou & Geng, 2004), (Turkan et al., 2004)). The explanation lies in the normalization procedure implied when constructing the ZEP feature.

Comparison with face feature fiducial points localization is not straightforward. While such methods localize significantly more points than simple eye centers localization, they also rely strongly on the inter-spatial relation among them to boost the overall performance. Furthermore, they often do not localize eye centers, but eye corners and the top/bottom of the eye, which in many cases are more stable than the eye center (i.e. not occluded or influenced by gaze). And yet we note that our method is comparable in terms of accuracy and significantly faster (if one normalizes the reported time by the number of detected points).

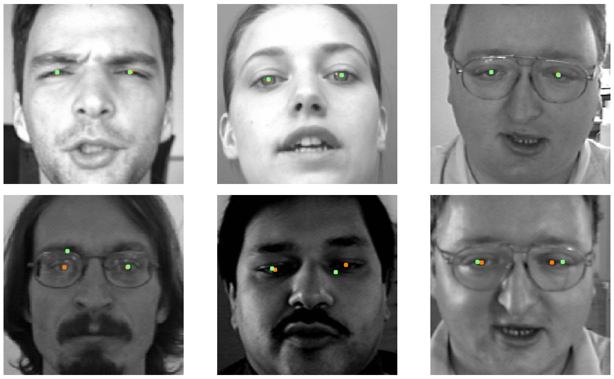


Figure 11. Face cropped images from BioID database. Top row shows images with eyes correctly localized, while bottom row shows failure cases.

Regarding other methods for eye localization, the proposed method ranks as one of the top methods for all accuracy tests, being always close to the best solution. Furthermore taking into account that on BioID database there are only ≈ 50 images (3%) with closed eyes, methods that search circular (symmetrical) shapes have better circumstances. Because we targeted images with expressions, we specifically included in our training data set closed eyes. To validate this assumption we tested with very good results on the Cohn-Kanade database showing that our method is more robust in that case as showed in figure 8 (b).

Considering as most important criterion the accuracy at $\epsilon < 0.05$, we note that 2011 and 2012 provide higher accuracy. Yet, we must also note and the highest performance achieved by a variation of the method described in (Valenti & Gevers, 2012), namely Val.&Gev.+SIFT contains a 10-fold testing scheme, thus using 9 parts of the BioID database for training. Furthermore, taking into account that BioID database was used for more that 10 years and provides limited variation, it has been concluded (Belhumeur et al., 2011), (Dantone et al., 2012) that other tests are also required to validate a method.

2012 provide results on other datasets and made public the associated code for their baseline system (Val.&Gev.+MS) which is not database dependent. 2011 do not provide results on any other database except BioID or source code, yet there is publicly available[2] code developed with author involvement. Thus, in continuation, we will compare the hereby proposed

[2] At `http://thume.ca/projects/2012/11/04/simple-accurate-eye-center-tracking-in-opencv/`.

Table 6. Comparison with state of the art (listed in chronological appearance) in terms of localization accuracy on the BioID database. The correct localization presents results reported by authors; values marked with "*" were inferred from authors plot. While 2004 reports only the value for $\varepsilon < 0.25$, the rest is reported by (Ciesla & Koziol, 2012). The method marked with † relied on a 10-fold training/testing scheme, thus, at a step, using 9 parts of the BioID database for training.

	Accuracy		
Method	$\epsilon < 0.05$	$\epsilon < 0.1$	$\epsilon < 0.25$
Proposed	**70.46**	**91.94**	**98.89**
(Jesorsky et al., 2000)	40.0	79.00	91.80
(Wu & Zhou, 2003)	10.0*	54.00*	93.00*
(Zhou & Geng, 2004)	47.7	74.5	97.9
(Cristinacce et al., 2004)	55.00*	96.00	98.00
(Campadelli et al., 2006)	62.00	85.20	96.10
(Hamouz et al., 2005)	59.00	77.00	93.00
(Turkan et al., 2004)	19.0*	73.68	99.46
(Kroon et al., 2008)	65.0	87.0	98.8
(Asteriadis et al., 2009)	62.0*	89.42	96.0
(Asadifard & Shanbezadeh, 2010)	47.0	86.0	96.0
(Timm & Barth, 2011)	82.50	93.40	98.00
(Valenti & Gevers, 2012)+MS	81.89	87.05	98.00
(Valenti & Gevers, 2012)+SIFT †	86.09	91.67	97.87
(Florea et al., 2012)	57.13	88.97	98.48

Table 7. Comparison with methods that localize a multitude (≈ 20) of points on the face on the BioID database. All the results were extrapolated from authors graphs. We note that in these cases eye localization was not the major objective and authors report an *average* error for the entire set of points; thus we also used *average* achieved error. The reported time is for determination of the entire set of points.

		Accuracy		**Time performance**		
Method	*No. pts.*	$\epsilon < 0.05$	$\epsilon < 0.1$	*Duration*	*Platform*	$\frac{Duration}{point}$
Proposed	**2**	**78.80**	**96.04**	**13 msec**	**i7 2,7 GHz**	**6.5 msec**
(Vukadinovic & Pantic, 2005)	20	15.0	78.0	n/a	n/a	n/a
(Milborrow & Nicolls, 2008)	17	66.0	95.0	230 msec	P4 3Ghz	13.53 msec
(Valstar et al., 2012)	22	74.00	95.00	n/a	n/a	n/a
(Belhumeur et al., 2011)	29	62.00	85.20	≈950 msec	i7 3.06GHz	32.75 msec
(Mostafa & Farag, 2012)	17	74.00	96.00	470 msec	i7 2.93GHz	27.64 msec

Table 8. Percentage of correct eye localization on the Cohn-Kanade database. We report results on the neutral poses, expression apex and overall and compare against method from (Valenti & Gevers, 2012), (Timm & Barth, 2011) and (Ding & Martinez, 2010). We marked with light-gray background the best achieved performance for each accuracy criterion and respectively for each image type.

	Accuracy			
Method	*Type*	$\epsilon < 0.05$	$\epsilon <$ 0[illegible]	[illegible]
Proposed	Neutral	**76.0**	**99.**[illegible]	[illegible]
	Apex	**71.9**	**95.**[illegible]	[illegible]
	Total	**73.9**	**97.3**	**100**
(Valenti & Gevers, 2012)	Neutral	46.0	95.7	99.6
	Apex	35.1	92.4	98.8
	Total	40.6	94.0	99.2
(Timm & Barth, 2011)	Neutral	66.0	95.4	99.0
	Apex	61.4	85.1	93.4
	Total	63.7	90.2	96.2
(Ding & Martinez, 2010)	Neutral	14.3	75.9	100
	Apex	11.8	72.8	100
	Total	13.1	74.4	100.

method against these two on other datasets. Additionally, we include the comparison against the eye detector developed by 2010 which has also been trained and tested on other database, thus is not BioID dependent.

5.5. Robustness to Eye Expressions

As mentioned in the introduction, we are specifically interested in the performance of the eye localization with respect to facial expressions, as real-life cases with fully opened eyes looking straight are rare. We tested the performance of the proposed method on the Cohn-Kanade database (Kanade et al., 2000). This database was developed for the study of emotions, contains frontal illuminated portraits and it is challenging through the fact that eyes are in various poses (near-closed, half-open, wide-open). We tested only on the neutral pose and on the expression apex image from each example. The correct eye locations, with standard precisions, are shown in table 8. Typical localization results are presented in figure 12, while the maximum, average and minimum errors are plotted in figure 8 (b).

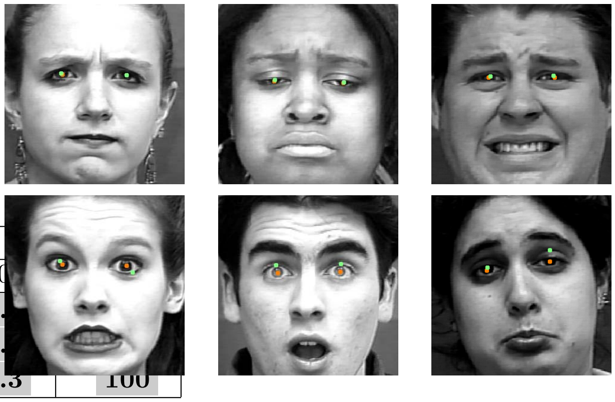


Figure 12. Face cropped images from the Cohn-Kanade database. The ground truth eyes are marked with red (dark grey), while detected eyes with green (light grey). Top row images show eyes correctly localized, while bottom row shows failure cases.

We note that solutions that try to fit a circular or a symmetrical shape over the iris, like (Valenti & Gevers, 2012) or (Timm & Barth, 2011), and thus, perform well on open eyes, do encounter significant problems when facing eyes in expressions (as it is shown in table 8). Taking into account the achieved results, which are comparable on neutral pose and expression apex images, we show that our method performs very well under such complex conditions. Achieved results indicate approximately a doubled accuracy when compared with the foremost state of the art method.

5.6. Robustness to Illumination and Pose

We systematically evaluated the robustness of the proposed algorithm with respect to lighting and pose changes. This was tested onto the Extended Yale Face Database B (B+) (Lee et al., 2005). We stress that part of the Yale B database (Georghiades et al., 2001) was used for training the MLP for lateral illumination, thus the training and testing sets are completely different.

The Extended Yale B database contains 16128 gray-scale images of 28 subjects, each seen under 576 viewing conditions (9 poses × 64 illuminations). The size of each image is 640×480. The robustness with respect to pose and with respect to illumination was evaluated separately.

For evaluating the robustness to illumination, we tested the system on 28 faces, in neutral pose, under changing illumination (64 cases). The results are summarized in table 9.

The system achieves reasonable results in the cases when even a human observer is not able to identify the eyes. As long as the illumination is constant over the eye, the system performs very well, proving the invariance to uniform illumination of the ZEP feature claim. Examples of localization while illumination varies are presented in figure 13.

For larger illumination angles, due to the uneven dis-

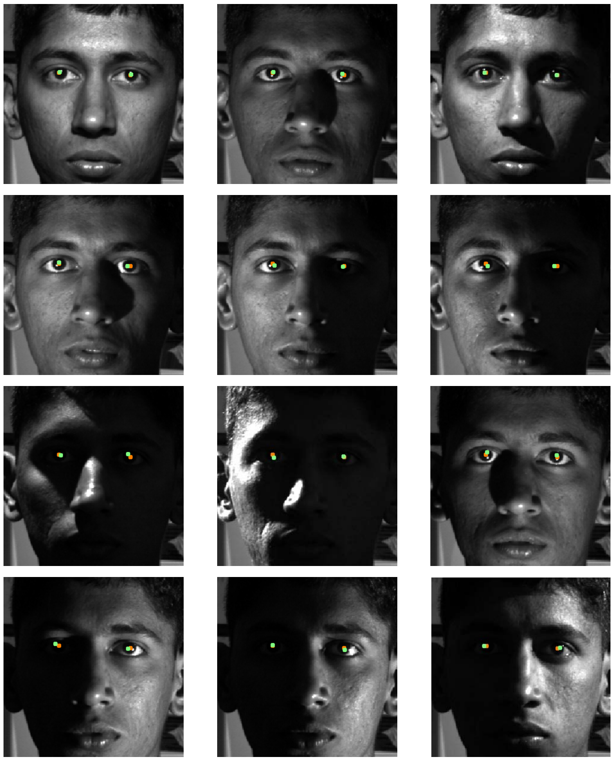
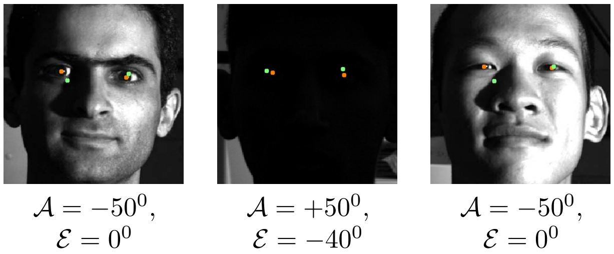


Figure 13. Face cropped images from the Extended YaleB database showing robustness to illumination.



$\mathcal{A} = -50^0$, $\mathcal{E} = 0^0$

$\mathcal{A} = +50^0$, $\mathcal{E} = -40^0$

$\mathcal{A} = -50^0$, $\mathcal{E} = 0^0$

Figure 14. Face cropped images from Extended Yale B database showing illumination cases that reveal limitations of the method. Specific shapes of projections caused by light and shade make the system prone to errors. Illumination angle is given by azimuth $\mathcal{A}$ and elevation $\mathcal{E}$.

Table 10. Pose variation studied on the Extended Yale B (B+) data set. The specified numerical values have been obtained for accuracy $\epsilon < 0.1$. For each category maximum two (corresponding to an accuracy of 92.86%) images were missed.

Azimuth Elevation	24^0	12^0	Frontal
Up	96.43	92.86	96.43
Neutral	96.43	96.43	92.86
Down	92.86	92.86	96.43

tribution of the shadows, the shape of the projections is significantly altered and the accuracy decreases. Examples, with cases where the shades are too strong or inopportunely placed and we reach lower results, are showed in figure 14.

To evaluate the robustness of the algorithm with respect to the face pose, we consider each of the 28 persons with frontal illumination, but under varying poses (9 poses for each person). Pose angles are in the set $\{0^0, 12^0, 24^0\}$, thus spanning the typical range for "frontal face". The results are shown in table 10 and visual examples in the figure 15. Taking into account that the maximum number of images that have the worst eye less accurate than 0.1 is 2, we may truthfully say that the proposed method is robust to face pose.

When compared with the method proposed in (Valenti & Gevers, 2012), our solution performs marginally better. If we consider the results reported in the mentioned paper, then the average result for accuracy at $\epsilon < 0.1$ is 88.07% computed on the smaller YaleB database (Georghiades et al., 2001) while our method reaches 89.85% on the same subset of azimuth and elevation illumination angles on the larger Extended YaleB database (Lee et al., 2005). If we compare the full results on the entire Extended YaleB database (including extreme illumination cases) then our method outperforms with small margin for high accuracies as shown in table 11. Our method performs significantly better than the ones proposed by 2011 and respectively 2010.

5.7. Accuracy in Real-Life Scenarios

While BioID, Cohn-Kanade and Extended YaleB databases include specific variations as they are acquired under controlled lighting conditions with only frontal faces, they cannot be considered too closely resembling real-life applications. In contrast, there are databases like the Labeled Face Parts in the Wild (LFPW) (Belhumeur et al., 2011) and the Labeled

Table 11. Comparative results on the Extended YaleB database

Method	$\epsilon < 0.05$	$\epsilon < 0.1$	$\epsilon < 0.25$
Proposed	*39.9*	*67.3*	*97.3*
(Valenti & Gevers, 2012)	37.8	66.6	98.5
(Timm & Barth, 2011)	20.1	34.5	51.5
(Ding & Martinez, 2010)	19.7	47.8	58.6

Table 9. Illumination variation studied on the Extended Yale B (B+) data set. The numerical values have been obtained for accuracy $\epsilon < 0.1$. If more cases were available in the given interval, the average is reported. If the specific case does not exist in the database, "n/a" is reported.

Azimuth / Elevation	$\pm[110^0 : 130^0]$	$\pm[70^0 : 90^0]$	$\pm[50^0 : 60^0]$	$\pm[20^0 : 35^0]$	$\pm[5^0 : 15^0]$	0^0
$-40^0 : -35^0$	n/a	73.21	58.93	60.71	n/a	92.86
$-20^0 : -10^0$	71.43	76.79	83.93	84.82	91.96	96.43
0^0	42.86	75.00	65.50	94.64	91.07	92.86
$10^0 : 20^0$	67.86	76.79	75.87	87.50	95.54	100
$40^0 : 45^0$	75.00	76.79	n/a	75.00	n/a	89.29
$65^0 : 90^0$	82.14	n/a	n/a	64.29	n/a	78.57

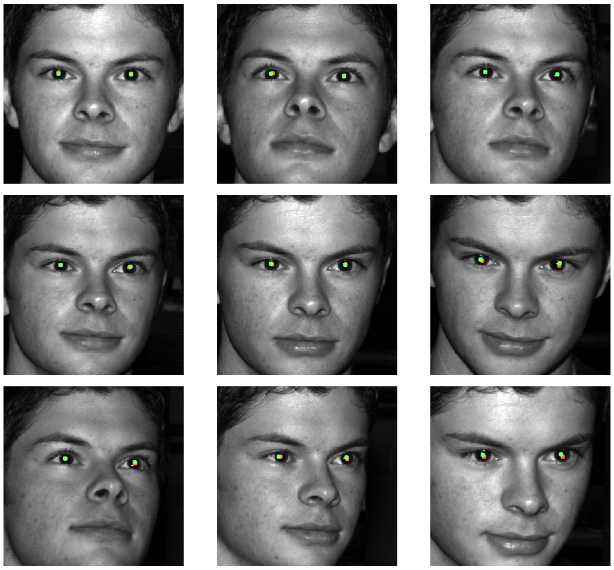


Figure 15. Face cropped images from Extended YaleB database showing robustness to pose.

Faces in the Wild (LFW) (Huang et al., 2007), which are randomly gathered from the Internet, contain large variations in the imaging conditions. While LFPW is annotated with facial point locations, only a subset of about 1500 images is made available and contains high resolution and rather qualitative images. In opposition, the LFW database contains more than 12000 facial images, having the resolution 250 × 250 pixels, with 5700 individuals that have been collected "in the wild" and vary in pose, lighting conditions, resolution, quality, expression, gender, race, occlusion and make-up.

The images difficulty is certified by the performance of human evaluation error as reported in (Dantone et al., 2012), which also provided annotations. While the ground truth is taken as the average of human markings for each point normalized to inter-ocular distance, human evaluation error is considered as the averaged displacement of the one marker.

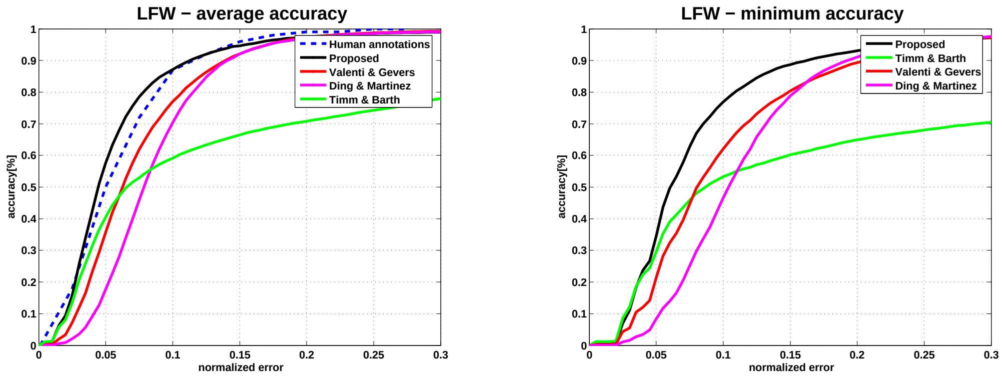
Examples of the results achieved on the LFW database may be seen in figure 16. Numerical results, compared with the solution from (Valenti & Gevers, 2012), (Timm & Barth, 2011), (Ding & Martinez, 2010) and with human evaluation error are presented in figure 17.

Regarding the achieved results, we note that even that our method was designed to work on large resolution faces, it provides accurate results when applied on smaller ones. As one can see in figure 17 we significantly outperform the state of the art solutions (Valenti & Gevers, 2012) and from (Timm & Barth, 2011) by almost 50% improvement at $\epsilon < 0.05$ accuracy, on an over 12000 image database that presents as close to real-life as possible cases and with more the method from (Ding & Martinez, 2010).

5.8. Algorithm Complexity

The entire algorithm requires only four divisions for the projections normalization and two for determination of the region weighting center with variable denominator per eye crop, and no high precision operations, therefore needing only limited fixed point precision. The ZEP+MLP combination is linear with respect to the size of scan eye rectangle $O(\delta N)$. The method was implemented in C around OpenCV functionality, on an Intel i7 at 2.7 GHz, on single thread and it takes 6 msec for both eyes on a face square of 300 × 300 pixels, which is a typical face size for HD - 720p (1280 × 720 pixels) format. We note that additional 7 msec are required for Face Detection.

The code can run in real-time while including face detection and further face expression analysis. Comparison with state of the art methods may be followed in the table 12 when comparing with other eye localization methods and on right hand side of table 7 when discussing face fiducial points localization solutions;



(a)

(b)

Figure 17. Results achieved on the LFW database: (a) Average accuracy and (b) minimum accuracy as imposed by the stringent criterion - eq. (9). With dashed blue line is the average error for human evaluation , black line – proposed method, red line – 2012, green line – 2011 and with magenta line – 2010

.

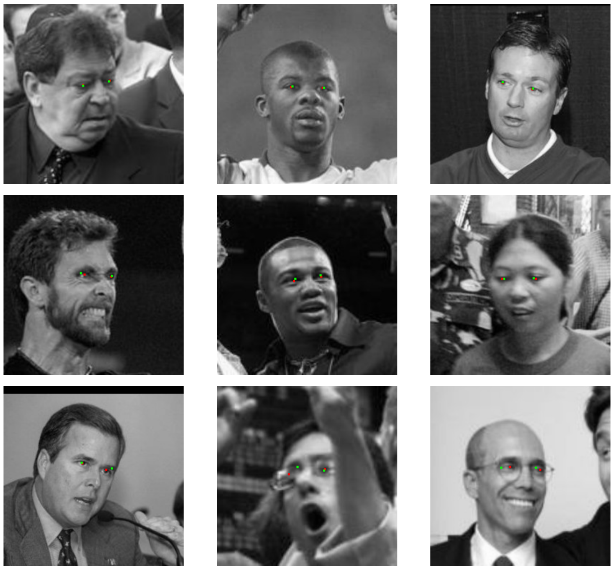


Figure 16. Images from Labelled Face in the Wild (LFW) database. The top row show images having accuracy higher than 0.05, medium row accuracy between 0.05 and 0.1 and bottom row accuracy lower than 0.1.

for some works the authors have not reported speed performance, but taking into account algorithm complexity, it is reasonable to presume that it is too large for real-time.

Trying to overcome the difficulty of comparison while different platforms were used for implementation, we rely as a unifying factor on the single thread benchmarking score provided by (PassMark Software Pty Ltd, retrieved January 2015) for specific CPU; this score will be denoted by CPU_s. It must be noted that such numbers should be considered with precaution since there do exist several CPUs that correspond to the description provided by authors (and we always took the best case) and the benchmark test may not be very relevant for the specific processing required by a solution.

To aggregate the overall time performance of a method we used the following formula:

$$T_p = \frac{fps \times \min\{M, N\}}{CPU_{power}} \quad (10)$$

where $M \times N$ is the frame size used for reported results. Note that the formula uses only one of the two dimensions that describe an image to cope with different aspect ratios.

The results for eye localization aggregated with the measure in equation (10) are showed in the table 12 when comparing with other eye localization methods. Our method rank second following the one proposed by 2000, but it gives consistently better results in terms of accuracy.

It has to be noticed that while, initially only 2012 reported comparable computational time, after integrating the larger frame size with processing power, our method turns to be 1.5 times faster. Furthermore, to be able to directly compare our computation time, we have modified the size of input face to be 120 × 120 which corresponds to a 320 × 240 pixels image, letting everything else the same and we find out that our method requires 1.6 msec to localize both eyes; given the additional 7 msec for face detection, we get a total time of 8.6 msec, that is equivalent with a frame rate of 116 frames/sec, proving that we clearly outperform

Table 12. Comparison with state of the art (listed in chronological appearance) in terms of time requirements. Regarding time performance,we compare ourself only with papers that provide some measure of duration. Also we note that, in general, authors reported non-optimized results on PC platforms and various image sizes. The reported time for (Asadifard & Shanbezadeh, 2010) is taken from (Ciesla & Koziol, 2012). With gray background we marked the best state of the art result. T_p is given by equation 10 and higher values are desired.

	Time performance				
	FrameRate	*Image size*	*Platform*	CPU_s	T_p
Proposed	**76 fps**	**1280× 720**	**i7 2.7 GHz**	**1747**	**31.23**
(Jesorsky et al., 2000)	33 fps	384 × 288	P3 850MHz	≈ 200	47.52
(Zhou & Geng, 2004)	15fps	320 × 240	Core 2	981	3.67
(Cristinacce et al., 2004)	0.7 fps	384 × 286	P3 500MHz	140	1.42
(Campadelli et al., 2006)	0.08 fps	384 × 286	P4 3.2GHz	720	0.032
(Hamouz et al., 2005)	0.07 fps	720 × 576	P4 2.8GHz	618	0.065
(Turkan et al., 2004)	12fps	384 × 286	n/a	n/a	n/a
(Kroon et al., 2008)	2fps	n/a	n/a	n/a	n/a
(Asteriadis et al., 2009)	3.84 fps	384 × 286	PM 1.6 GHz	514	2.13
(Asadifard & Shanbezadeh, 2010)	15 fps	320 × 240	Core 2	981	3.67
(Valenti & Gevers, 2012)+MS	90 fps	320 × 240	Core2 2.4GHz	981	22.01
(Valenti & Gevers, 2012)+SIFT	29 fps	320 × 240	Core2 2.4GHz	981	7.09
(Florea et al., 2012)					

the method from (Valenti & Gevers, 2012).

5.9. Discussion

The previous subsections within this "Results and discussions" part have guided through various experiments and measurements that present a through comparison of eye localization performance of the here proposed ZEP eye localization method, which is shown to perform remarkably well among a wide variety of conditions and datasets. Some of the presented numbers and experiments deserve yet a supplemental emphasis and clarification.

A first issue of discussion is related to the experimental setup, namely the databases that are currently used in the assessment of algorithms accuracy. BioID has gained through the years widespread recognition, as it was one of the earliest face image databases that contain facial landmark ground truth annotations. As such, BioID was intensively used for accuracy comparisons, with a clear tendency over time to concentrate the efforts in getting top results on BioID alone. As one may have noticed in table 6, the here proposed method is outperformed on BioID by the algorithms proposed by 2011 and 2012.

We can notice that most the methods are overtrained in standard conditions, and thus perform very well within their over–learned domain. As such, we claim that these approaches are not relevant in a broader, real-life testing scenario. The approaches proposed by 2012 and 2011 are thus retained as a significant eye location methods and we further tested them; we also included the solution from (Ding & Martinez, 2010) as being a high profile method build outside the BioID database; the results showed that we outperform these methods by a gross margin.

As anyone knowledgeable in the field observes, the BioID database contains mostly frontal pose, frontal illumination and neutral expression faces, and catches only a small glimpse of the problems related to eye localization. As such, intensive performance comparison must be realized outside these standard conditions, as 2012 does in the case of varying illumination and pose and we do in the case of noise, variable illumination, expression and pose variations. Several tests that are reviewed again here prove the superior performance of the proposed ZEP eye localization method in these extreme conditions.

The non-frontal illumination and the subject pose variations are key issues in real-life, unconstrained applications. Typically these are tested within the Extended YaleB database, where ZEP performs marginally better (+2%) than method in (Valenti & Gevers, 2012) and significantly better than (Timm & Barth, 2011) and (Ding & Martinez, 2010), as shown in Table 11. Subject emotional expressions hugely affect eye shape and surroundings. The Cohn-Kanade database is the state of the art testbed in emotion-related tasks; in this case, the ZEP eye localization outperforms (Timm & Barth, 2011) by 10%, (Valenti & Gevers, 2012) by some 30%, and (Ding & Martinez, 2010) by 60% as shown in table 8.

Within all databases, closed eyes present an independent challenge. As noticed by the authors in (Valenti & Gevers, 2012), their method is prone to errors in detecting the closed eye center (which is confirmed by the experiments across all databases). The proposed ZEP method is much more robust to closed eyes, due to the way in which the eye profile is described within the proposed encoding of the luminance profiles.

Finally, we consider that the most relevant test is performed on the LFW database, taking into account the size (more than 12000 images), image resolution (extremely low) and especially the fact that images were acquired "in the wild". Yet, on the LFW database, which is currently one of the most challenging tasks, we outperform the method in (Timm & Barth, 2011) by at least 5% the one in (Valenti & Gevers, 2012) by a gross margin (+13%) and respectively the method form (Ding & Martinez, 2010) by near 30%. Nonetheless we much closer to human accuracy (as shown in figure 17 (a)).

Regarding the computational complexity, the here proposed method requires a computational time which is inferior to the time required by the method from (Jesorsky et al., 2000); yet the accuracy of the here proposed method is significantly higher. If we compare only the computation time, without considering the image size, one may consider the method from (Valenti & Gevers, 2012) to be faster. Yet, tests showed that the here proposed solution is still faster than the implementation from (Valenti & Gevers, 2012) at equal image resolution (namely 320 × 240). We thus claim that the here proposed method is the fastest solution from the select group of high accuracy methods.

6. Conclusions

In this paper, we proposed a new method to estimate the eye centers location using a combination of Zero-based Encoded image Projections and a MLP. The eye location is determined by discriminating between eyes and non-eyes by analyzing of the normalized image projections encoded with the zero-crossing based method. The extensive evaluation of the proposed approach showed that it can achieve real-time high accuracy. While the ZEP feature was used for eye description, we consider that it is general-enough and may be used in numerous problems.

References

Asadifard, M. and Shanbezadeh, J. Automatic adaptive center pupil detection using face detection and cdf analysis. In *IMECS*, pp. 130–133, 2010.

Asteriadis, Stylianos, Nikolaidis, Nikos, and Pitas, Ioannis. Facial feature detection using distance vector fields. *Pattern Recognition*, 42(7):1388 – 1398, 2009.

Becker, H. C., Nettleton, W. J., Meyers, P. H., Sweeney, J. W., and Nice, C. M. Digital computer determination of a medical diagnostic index directly from chest X-ray images. *IEEE Trans. on Biomedical Engineering*, 1964(3):62 – 72, July 11.

Belhumeur, P.N., Jacobs, D.W., Kriegman, D.J., and Kumar, N. Localizing parts of faces using a consensus of exemplars. In *IEEE CVPR*, pp. 545 – 552, 2011.

Blelloch, G. E. Prefix sums and their applications. synthesis of parallel algorithms. Technical report, University of Massachusetts, 1990.

Campadelli, P., Lanzarotti, R., and Lipori, G. Precise eye localization through a general-to-specific model definition. In *BMVC*, pp. I – 187, 2006.

Ciesla, M. and Koziol, P. Eye pupil location using webcam. *Human-Computer Interaction*, 2012.

Cootes, T. F., Edwards, G. J., and Taylor, C. J. Active appearance models. *IEEE Transactions on PAMI*, 23(6):681 – 685, 2001.

Cristinacce, D., Cootes, T., and Scott, I. A multi-stage approach to facial feature detection. In *BMVC*, pp. 277 – 286, 2004.

Crow, F. Summed-area tables for texture mapping. *Proceedings of SIGGRAPH*, 18(3):207 – 212, 1984.

Dantone, M., Gall, J., Fanelli, G., and Gool, L. Van. Real-time facial feature detection using conditional regression forests. In *IEEE CVPR*, pp. 2578 – 2585, 2012.

Ding, Liya and Martinez, Aleix M. Features versus context: An approach for precise and detailed detection and delineation of faces and facial features. *IEEE Trans Pattern Anal Mach Intell.*, 32(11):2022 – 2038, 2010.

Everingham, Mark and Zisserman, Andrew. Regression and classification approaches to eye localization in face images. In *IEEE FGR*, pp. 441 – 446, 2006.

Feng, G. C. and Yuen, P. C. Variance projection function and its application to eye detection for human face recognition. *Pattern Recognition Letters*, 19(9): 899 – 906, July 1998.

Florea, Laura, Florea, Corneliu, Vertan, Constantin, and Vranceânu, Ruxandra. Zero-crossing base image projections encoding for eye localization. In *EUSIPCO*, pp. 150 – 154, 2012.

Georghiades, A., Belhumeur, P., and Kriegman, D. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. on PAMI*, 23(6):643 – 660, 2001.

Gonzalez, Rafael and Woods, Richard E. *Digital Image Processing*. Prentice Hall, New Jersey, 2-nd edition, 2001.

Gonzalez-Ortega, D., Daz-Pernas, F. J., Anton-Rodrguez, M., Martnez-Zarzuela, M., and Dez-Higuera, J. F. Real-time vision-based eye state detection for driver alertness monitoring. *Pattern Analysis and Applications*, 16(3):285 – 306, 2013.

Hamouz, M., Kittlerand, J., Kamarainen, J. K., Paalanen, P., Kalviainen, H., and Matas, J. Feature-based affine-invariant localization of faces. *IEEE Trans. on PAMI*, 27(9):643 – 660, 2005.

Hansen, Dan Witzner and Ji, Qiang. In the eye of the beholder: A survey of models for eyes and gaze. *IEEE Trans. on PAMI*, 32(3):478 – 500, March 2010.

Huang, G., Ramesh, M., Berg, T., and Learned-Miller, E. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report, University of Massachusetts, 2007.

Jesorsky, O., Kirchberg, K., and Frischolz, R. Robust face detection using the Hausdorff distance. In Bigun, J. and Smeraldi, F. (eds.), *AVBPA*, pp. 90–95. Springer, 2000.

Jones, M. and Viola, P. Fast multi-view face detection. Technical Report 096, Mitsubishi Electric Research Laboratories, 2003.

Kanade, T. Picture processing by computer complex and recognition of human faces. Technical Report, Kyoto University, Department of Information Science, 1973.

Kanade, T., Cohn, J. F., and Tian, Y. Comprehensive database for facial expression analysis. In *IEEE FG*, pp. 46–53, 2000.

Kim, S., Chung, S.-T., Jung, S., Oh, D., Kim, J., and Cho., S. Multi-scale Gabor feature based eye localization. In *WASET*, volume 21, pp. 483–487, 2007.

King, R. A. and Phipps, T. C. Shannon, TESPAR and approximation strategies. *Computers & Security*, 18 (5):445 – 453, 1999.

Kroon, B., Hanjalic, A., and Maas, S. M. Eye localization for face matching: is it always useful and under what conditions. In *CIVR*, pp. 379 – 387, 2008.

Lee, K.C., Ho, J., and Kriegman, D. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Trans. PAMI*, 27(5):684–698, 2005.

Liu, Weifeng, Wang, Yanjiang, and Jia, Lu. An effective eye states detection method based on projection. In *ICSP*, pp. 829 – 831, 2010.

Milborrow, S. and Nicolls, F. Locating facial features with an extended active shape model. In *ECCV*, pp. 504 – 513, 2008.

Mostafa, Eslam and Farag, Aly. Complex bingham distribution for facial feature detection. In *ECCV, Workshops and Demonstrations*, pp. 330–339, 2012.

Nefian, Ara V. and Hayes, Monson H. Maximum likelihood training of the embedded HMM for face detection and recognition. In *IEEE ICIP*, pp. 33 – 36, 2000.

Niu, Z., Shan, S., Yan, S., Chen, X., and Gao, W. 2D cascaded adaboost for eye localization. In *IEEE ICPR*, pp. 1216 – 1219, 2006.

PassMark Software Pty Ltd. CPU Mark - high–mid range, retrieved January 2015. from `http://www.cpubenchmark.net/singleThread.html`.

Ramirez, G.A. and Fuentes, O. Multi-pose face detection with asymmetric haar features. In *IEEE Workshop on Applications of Computer Vision, WACV 2008*, pp. 1 – 6, 2008.

Timm, F. and Barth, E. Accurate eye centre localisation by means of gradients. In *VISAPP*, pp. 125–130, 2011.

Turkan, Mehmet, Pardas, Montse, and Cetin, A. Enis. Edge projections for eye localization. *Optical Engineering*, 47:047007, 2004.

Valenti, R. and Gevers, T. Accurate eye center location and tracking using isophote curvature. In *IEEE CVPR*, pp. 1–8, 2008.

Valenti, R. and Gevers, T. Accurate eye center location through invariant isocentric patterns. *IEEE Trans. PAMI*, 34(9):1785–1798, 2012.

Valstar, M., Martinez, B., Binefa, X., and Pantic, M. Facial point detection using boosted regression and graph models. In *IEEE CVPR*, pp. 27292736, 2012.

Verjak, M. and Stephancic, M. An anthropological model for automatic recognition of the male human face. *Ann. Human Biology*, 21:363–380, 1994.

Viola, P. and Jones, M. Robust real-time face detection. *International Journal of Computer Vision*, 57 (2):137–154, 2004.

Vukadinovic, D. and Pantic, M. Fully automatic facial feature point detection using gabor feature based boosted classifiers. In *IEEE Conf. on Systems, Man and Cybernetics*, pp. 1692 – 1698, 2005.

Wu, J. and Zhou, Zhi-Hua. Efficient face candidates selector for face detection. *Pattern Recognition*, 36 (5):1175–1186, 2003.

Zhou, Zhi-Hua and Geng, Xin. Projection functions for eye detection. *Pattern Recognition*, 37(5):1049 – 1056, 2004.